

KARTA OPISU MODUŁU KSZTAŁCENIA		
Nazwa modułu/przedmiotu Biznesowe systemy rozproszone		Kod 1010512321010510069
Kierunek studiów Informatyka	Profil kształcenia (ogólnoakademicki, praktyczny) ogólnoakademicki	Rok / Semestr 1 / 2
Ścieżka obieralności/specjalność Technologie wytwarzania oprogramowania	Przedmiot oferowany w języku: polski	Kurs (obligatoryjny/obieralny) obligatoryjny
Stopień studiów: II stopień	Forma studiów (stacjonarna/niestacjonarna) stacjonarna	
Godziny Wykłady: 30 Ćwiczenia: - Laboratoria: 30 Projekty/seminaria: -		Liczba punktów 4
Status przedmiotu w programie studiów (podstawowy, kierunkowy, inny) kierunkowy		(ogólnouczelniany, z innego kierunku) z danego kierunku
Obszar(y) kształcenia i dziedzina(y) nauki i sztuki nauki techniczne		Podział ECTS (liczba i %) 4 100%
Odpowiedzialny za przedmiot / wykładowca:		
<p>dr inż. Tomasz Pawlak email: Tomasz.Pawlak@cs.put.poznan.pl tel. 61 6653022 Instytut Informatyki ul. Piotrowo 2, 60-965 Poznań</p>		
Wymagania wstępne w zakresie wiedzy, umiejętności, kompetencji społecznych:		
1	Wiedza:	Efekty kształcenia ze studiów I stopnia zdefiniowane w Uchwale Senatu PP, a szczególnie efekty K_W1-2, K_W4, K_W6-15, weryfikowane w procesie rekrutacji na studia 2 stopnia ? efekty te prezentowane są w serwisie internetowym wydziału www.fc.put.poznan.pl
2	Umiejętności:	Efekty kształcenia ze studiów I stopnia zdefiniowane w Uchwale Senatu PP, a szczególnie efekty K_U1-2, K_U4, K_U7-8, K_U14-20, K_U22-23, K_U26, weryfikowane w procesie rekrutacji na studia 2 stopnia ? efekty te prezentowane są w serwisie internetowym wydziału www.fc.put.poznan.pl oraz umiejętność pozyskiwania informacji ze wskazanych źródeł.
3	Kompetencje społeczne	Efekty kształcenia ze studiów I stopnia zdefiniowane w Uchwale Senatu PP, a szczególnie efekty K_K1-9, weryfikowane w procesie rekrutacji na studia 2 stopnia ? efekty te prezentowane są w serwisie internetowym wydziału www.fc.put.poznan.pl Powinien również rozumieć konieczność poszerzania swoich kompetencji i mieć gotowość do podjęcia współpracy w ramach zespołu. Ponadto w zakresie kompetencji społecznych student musi prezentować takie postawy jak uczciwość, odpowiedzialność, wytrwałość, ciekawość poznawcza, kreatywność, kultura osobista, szacunek dla innych ludzi.
Cel przedmiotu:		
<ol style="list-style-type: none"> Przekazanie studentom podstawowej wiedzy dotyczącej wykorzystania narzędzi programistycznych, w zakresie wytwarzania i integracji systemów rozproszonych stosowanych w biznesie. Rozwijanie u studentów umiejętności rozwiązywania problemów dotyczących analizy i projektowania systemów rozproszonych w ramach technologii stosowanych w biznesie. Omówienie zagadnień związanych z bezpieczeństwem i wydajnością systemów rozproszonych stosowanych w biznesie. Kształtowanie u studentów umiejętności pracy zespołowej w ramach ćwiczeń laboratoryjnych w trakcie wykonywania projektów technicznych w grupach. 		
Efekty kształcenia i odniesienie do kierunkowych efektów kształcenia		
Wiedza:		
<ol style="list-style-type: none"> ma uporządkowaną, podbudowaną teoretycznie wiedzę ogólną w zakresie architektury systemów komputerowych i technologii służących do integracji i implementacji systemów informatycznych w środowisku rozproszonym. - [K_W4] ma podbudowaną teoretycznie szczegółową wiedzę związaną z wybranymi zagadnieniami z zakresu informatyki, takimi jak: zaprojektowanie, wytworzenie i testowanie systemu rozproszonego. - [K_W5] ma wiedzę o trendach rozwojowych i najistotniejszych nowych osiągnięciach w informatyce w zakresie technologii internetowych - [K_W6] ma podstawową wiedzę o cyklu życia systemów informatycznych sprzętowych lub programowych - [K_W7] zna podstawowe metody, techniki i narzędzia stosowane przy rozwiązywaniu złożonych zadań inżynierskich z wybranego obszaru informatyki - [K_W8] 		

Umiejętności:

1. potrafi pozyskiwać informacje z literatury, dokonywać ich interpretacji i krytycznej oceny, wyciągać wnioski - [K_U1]
2. potrafi określić kierunki dalszego uczenia się i zrealizować proces samokształcenia - [K_U5]
3. potrafi - przy formułowaniu i rozwiązywaniu zadań inżynierskich - integrować wiedzę z różnych obszarów informatyki oraz zastosować podejście systemowe, uwzględniające także aspekty pozatechniczne - [K_U10]
4. potrafi formułować i testować hipotezy związane z problemami inżynierskimi - [K_U12]
5. potrafi ocenić przydatność i możliwość wykorzystania nowych osiągnięć (metod i narzędzi) oraz nowych produktów informatycznych - [K_U13]
6. potrafi sformułować specyfikację funkcjonalną w formie przypadków użycia - [K_U22]
7. potrafi sformułować wymagania pozafunkcjonalne dla wybranych charakterystyk jakościowych - [K_U23]
8. potrafi ocenić przydatność metod i narzędzi służących do rozwiązania zadania inżynierskiego, polegającego na budowie lub ocenie systemu informatycznego lub jego składowych, w tym dostrzec ograniczenia tych metod i narzędzi - [K_U24]
9. potrafi wybrać język programowania odpowiedni do danego zadania programistycznego - [K_U26]
10. potrafi zaprojektować, zgodnie z zadaną specyfikacją, uwzględniającą aspekty pozatechniczne, system informatyczny, co najmniej w części, używając właściwych metod, technik i narzędzi, w tym przystosowując do tego celu istniejące lub opracowując nowe narzędzia - [K_U27]

Kompetencje społeczne:

1. rozumie, że w informatyce wiedza i umiejętności bardzo szybko stają się przestarzałe - [K_K1]
2. rozumie przyczyny wadliwie działających systemów informatycznych, które doprowadziły do poważnych strat finansowych, społecznych lub też do poważnej utraty zdrowia, a nawet życia - [K_K4]
3. potrafi odpowiednio określić priorytety służące realizacji określonego przez siebie lub innych zadania - [K_K6]
4. potrafi współdziałać i pracować w grupie, przyjmując w niej różne role - [K_K5]

Sposoby sprawdzenia efektów kształcenia

Efekty kształcenia przedstawione wyżej weryfikowane są w następujący sposób:

Ocena formująca:

- a) w zakresie wykładów:
 - na podstawie odpowiedzi na pytania dotyczące materiału omówionego na wykładach,
- b) w zakresie laboratoriów:
 - na podstawie oceny bieżącego postępu realizacji zadań laboratoryjnych,

Ocena podsumowująca:

- a) w zakresie wykładów weryfikowanie założonych efektów kształcenia realizowane jest przez:
 - ocenę wiedzy i umiejętności wykazanych na egzaminie pisemnym,

Egzamin składa się ze:

- Zbioru od 8 do 12 pytań zamkniętych, z których na każde pytanie można udzielić jedną prawidłową odpowiedź z czterech możliwych. Za każdą poprawną odpowiedź uzyskuje się 1 punkt, a za błędną odejmowana jest 1/3 punktu.
- Zbioru od 3 do 5 pytań otwartych, za które można uzyskać od 2 do 3 punktów.

- b) w zakresie laboratoriów weryfikowanie założonych efektów kształcenia realizowane jest przez:

- ocenę umiejętności związanych z realizacją ćwiczeń laboratoryjnych obejmujących programowanie,
- ocenę prezentacji przygotowywanej częściowo w trakcie zajęć, a częściowo w ramach pracy domowej; ocena ta obejmuje także umiejętność pracy w zespole.

Uzyskiwanie punktów dodatkowych za aktywność podczas zajęć, a szczególnie za:

- omówienia dodatkowych aspektów zagadnienia,
- efektywność zastosowania zdobytej wiedzy podczas rozwiązywania zadanego problemu,
- umiejętność współpracy w ramach zespołu praktycznie realizującego zadanie szczegółowe w laboratorium,
- uwagi związane z udoskonaleniem materiałów dydaktycznych,
- prezentacje autoreferatu powiązanego z tematem zajęć.

Treści programowe

Program wykładu obejmuje następujące zagadnienia:

Wprowadzenie do biznesowych systemów rozproszonych - Wprowadzone zostaną podstawowe pojęcia oraz zakres tematyczny wykładu. Wskazane zostaną zalety przetwarzania rozproszonego i różnice między przetwarzaniem rozproszonym oraz równoległym. Ogólnie zostaną przedstawione najważniejsze pojęcia dotyczące jakości przetwarzania i wydajności w środowisku rozproszonym. Przedstawione zostaną przykłady zastosowania systemów rozproszonych.

Middleware w sieciach lokalnych ? Przedstawione i scharakteryzowane zostaną rodzaje platform middleware stosowane w sieciach korporacyjnych przedsiębiorstwa, ze szczególnym nastawieniem na platformy budowy aplikacji. Zostaną omówione technologie RPC, monitorów transakcyjnych, brokerów obiektowych, monitorów obiektowych i kolejek wiadomości.

Integracja platform biznesowych i middleware w sieciach rozległych ? W zakresie integracji platform biznesowych przedstawione i szczegółowo omówione zostanie środowisko typu Enterprise Application Integration (EAI) oraz jego szczególne realizacje pod postacią brokerów wiadomości i systemów workflow. W zakresie middleware dla sieci rozległych zostaną omówione technologie Java Enterprise Edition oraz Microsoft .NET Framework, a także technologie i formaty wymiany wiadomości przez Internet: General Inter-ORB Protocol, EDIFACT, XML, JSON.

Modelowanie systemu rozproszonego przy pomocy UML - przedstawione zostaną podstawowe diagramy UML oraz przykłady ich zastosowania w kontekście modelowania systemu rozproszonego. Omówione zostaną elementy notacji UML, które mogą zostać wykorzystane do modelowania systemu rozproszonego. Przedstawiony zostanie plan, według którego należy podejść do modelowania systemu rozproszonego w języku UML.

Usługi internetowe ? Zostanie wprowadzone i scharakteryzowane pojęcie usługi internetowej oraz wyjaśnione zostaną podstawowe różnice między usługami internetowymi, a middleware. Wprowadzone i scharakteryzowane zostaną podstawowe standardy usług internetowych: Simple Object Access Protocol, Web Service Description Language, Universal Description, Discovery and Integration.

Protokoły koordynacji usług ? Wprowadzone i scharakteryzowane zostaną protokoły WS-Coordination, WS-Transaction, WS-ReliableMessaging, ebXML, RosettaNet.

Kompozycja usług ? Wprowadzone zostanie pojęcie kompozycji usług internetowych, z rozróżnieniem na model kompozycji typu choreografia oraz model typu orkiestracja. Zostaną omówione związki kompozycji usług z koordynacją wykonania usług. Omówiony zostanie standard Web Services Business Process Execution Language służący kompozycji usług internetowych.

Analiza systemów biznesowych ? Omówione zostaną techniki analizy statycznej systemów poprzez analizę ich kodu źródłowego lub wykonywalnego, a także analizy dynamicznej działania systemu. Zostaną omówione uwarunkowania prawne tego typu działań, oraz narzędzia ich realizacji, w tym: deasemblerzy i dekompiletory kodu, debuggery i narzędzia do wstrzykiwania kodu, narzędzia analizy zdarzeń generowanych przez działający system.

Zajęcia laboratoryjne prowadzone są w formie piętnastu 2-godzinnych ćwiczeń, odbywających się w laboratorium, poprzedzonych sesją instruktażową na początku semestru. Ćwiczenia realizowane są samodzielnie lub przez 2-osobowe zespoły studentów. Program laboratorium obejmuje następujące zagadnienia:

Ćwiczenia z technologią Windows Communication Foundation

Ćwiczenia z technologią usług REST w NodeJS

Ćwiczenia z technologią JSON-RPC

Ćwiczenia z technologią JSON-WSP

Ćwiczenia z technologią XML-RPC

Ćwiczenia z technologią CORBA

Ćwiczenia z technologią Java RMI

Ćwiczenia z technologią Java Message Service (ApacheMQ)

Ćwiczenia z technologią Apache Kafka

Realizacja projektu? zaprojektowania, implementacja i przetestowanie prostego systemu rozproszonego wraz z implementacją własnego modułu komunikacyjnego

Metody dydaktyczne:

1. wykład: prezentacja multimedialna.
2. ćwiczenia laboratoryjne: rozwiązywanie zadań, ćwiczenia praktyczne, praca w zespole, demonstracja.

Literatura podstawowa:

1. Gustavo Alonso, Fabio Casati, Harumi Kuno, Vijay Machiraju, Web Services: Concepts, Architectures and Applications, Springer, 2004

Literatura uzupełniająca:		
1. Dokumentacja standardów technicznych WWW: http://www.w3.org		
2. Dokumentacja UML: http://www.omg.org		
3. Standardy związane z technologią JEE: http://www.jcp.org		
4. Gerald Brose, Andreas Vogel, Keith Duddy: Java programming with Corba: advanced techniques for building distributed applications, Wiley, 1997		
5. Alur, Deepak., Crupi, John., Malks, Dan., Jońca, Rafał. Tł.: J2EE : wzorce projektowe : katalog wzorców projektowych J2EE z Java Sun Center, wzorce i strategie projektowe dla J2EE i usług sieciowych, refaktoryzacja i niezalecane techniki, wzorce, strategie i refaktoryzacja kodu źródłowego, Helion, 2004		
6. Arnon Rotem-Gal-Oz: SOA Patterns, Manning Publications, 2012		
7. David Dossot, John D'Emic, Victor Romero: Mule in Action, Manning Publications, 2014		
Bilans nakładu pracy przeciętnego studenta		
Czynność	Czas (godz.)	
1. udział w zajęciach laboratoryjnych	30	
2. przygotowanie do ćwiczeń laboratoryjnych	8	
3. wykonanie (w ramach pracy własnej) prezentacji technologii	6	
4. udział w konsultacjach związanych z realizacją procesu kształcenia, w szczególności ćwiczeń laboratoryjnych / projektu (częściowo mogą być realizowane drogą elektroniczną)	6	
5. udział w wykładach	30	
6. napisanie programów, uruchomienie i weryfikacja (czas poza zajęciami laboratoryjnymi)	6	
7. zapoznanie się ze wskazaną literaturą / materiałami dydaktycznymi	6	
8. omówienie wyników egzaminu	2	
9. przygotowanie do egzaminu i udział w nim (2 godz.)	6	
Obciążenie pracą studenta		
forma aktywności	godzin	ECTS
Łączny nakład pracy	100	4
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	68	3
Zajęcia o charakterze praktycznym	50	2